

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-358064

(43)Date of publication of application : 26.12.2000

(51)Int.Cl.

H04L 12/56

(21)Application number : 11-170479

(71)Applicant : NIPPON TELEGR & TELEPH
CORP <NTT>

(22)Date of filing : 17.06.1999

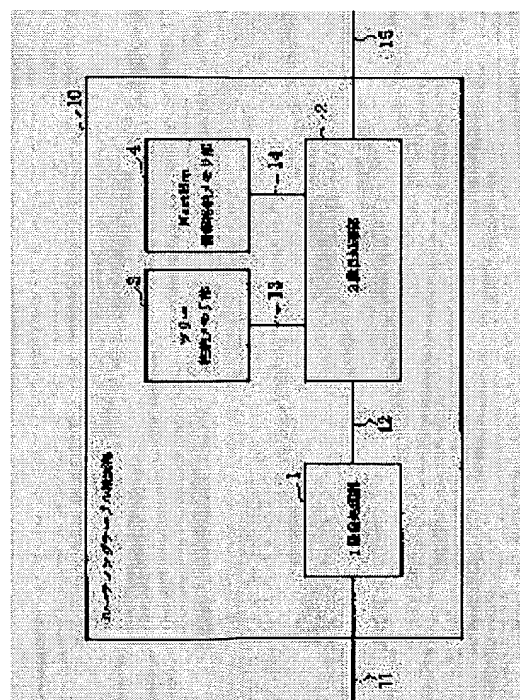
(72)Inventor : UGA MASANORI
SHIOMOTO KOHEI

(54) DEVICE AND METHOD FOR ROUTING TABLE RETRIEVAL

(57)Abstract:

PROBLEM TO BE SOLVED: To enable fast retrieval by reducing the frequency and memory capacity of routing table retrieval.

SOLUTION: Retrieval is carried out in two stages; and 1st retrieval is performed by a 1st-stage process part 1 by using CAM and 2nd retrieval is performed by arranging a tree at a 2nd-stage process part 2. All Prefix's entering a table are divided into a range determined by Prefix length and Prefix's generated by shortening the Prefix's to the minimum Prefix length in the range are stored in the CAM. The 1st-stage process part 1 compares an input IP address with the Prefix's stored in the CAM and outputs the address of a tree storage memory part 3 corresponding to a matching Prefix. The process moves to the node corresponding to the output of the CAM halfway in a tree, retrieval is carried out in the limited range, and at the end of the retrieval, information in a NextHop information storage memory 4 is outputted as a final result.



LEGAL STATUS

[Date of request for examination]

16.10.2001

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

* NOTICES *

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] About a high speed and the mass routing table searching method, especially this invention can restrict the count of retrieval, and relates to the routing searching method which can search routing table at a high speed.

[0002]

[Description of the Prior Art] The routing table searching method using the conventional CAM (Content Addressable Memory) is explained. Drawing 2 is the explanatory view of the conventional routing table searching method for having used CAM of operation. As shown in drawing 2, CAM 16-1 to 16-25 is prepared for every Prefix length, and Prefix on routing table is stored in CAM which corresponds with the Prefix length. That is, if it is Prefix of the Prefix length 8, it stores in CAM(8)16-1, if it is Prefix of the Prefix length 9, it stores in CAM(9)16-2, and if it is Prefix of the Prefix length 32 similarly, it stores in CAM(32) 16-25. In case it searches, the destination IP address which should be searched is inputted into all CAM 16-1 to 16-25. CAM is Next, when the destination IP address inputted into CAM corresponding to the Prefix length p compares a part for p bits of high orders of a destination IP address with Prefix which CAM stores and it is in agreement. Information or Next of Hop It had the information on Hop, and also the Index information on a table is outputted. In addition, it is Priority when in agreement by two or more CAM. The output from CAM which stores longest Prefix among Encoder(s) 17 is chosen. By the method of drawing 2, the Prefix in routing table itself all had to be stored in CAM, consequently there was a problem that many amounts of memory were needed.

[0003] Drawing 3 is the explanatory view of the conventional routing table searching method for having used the PATORISHIA tree. By the searching method shown in drawing 3, each nodes 31-34 have the Prefix length of Prefix and its node, respectively. When the IP address which should be searched to a certain node reaches, even the high order Prefix length bit eye of Prefix and a destination IP address which the node shows is compared. Next which the table corresponding to the node which was in agreement immediately before has in not being in agreement as a result of a comparison Next which ends retrieval as information which searches for the information on Hop, and the table corresponding to the node has when in agreement The information on Hop is read and recorded. Next, Next which moved to the child node with reference to the 1st [+] bit of the high order Prefix length of a destination IP address when a child node existed with the value, continued retrieval, ended retrieval when a child node did not exist on the other hand, and was recorded It considers as the information which searches for the information on Hop. drawing 3 -- 10000101 -- since the value of the 1st bit of the high order of a destination IP address branches by the root node 31 first 0 or 1, it moves to a node (S) 32 since a check bit (the 1st bit of the high order of a destination IP address) is 1, and a part for a high order triplet is compared and it is in agreement when the destination IP address ... is inputted, the 4th bit which is a check bit is investigated. Since a check bit is 0, it moves to a node (U) 33. Here, since a part for 5 bits of high orders is compared and it is not in agreement, retrieval is ended and let 100/3 of the nodes (S) 32 which were in agreement just before be an answer. By the method of drawing 3, when an increment and

address length of the number of entries of routing table became long, the wooden depth became deep, and the count of retrieval increased in connection with it, consequently there was a problem that retrieval took time amount.

[0004] Drawing 4 is the explanatory view of the routing table searching method for having used the Multiway tree. This method is the purpose which reduces the count of access to a node in the case of retrieval, and is the approach of making the wooden depth shallow by giving two or more three or more child nodes to one parent node. That is, the wooden depth can be made shallow by making it the tree made to deform like drawing 4 (a) to drawing 4 (b). The effective node (they are A, B, E, F, and D all over drawing) which does not exist on the stage (Prefix length is the stage of the multiple of 3) which drawing 4 (b) is an example ($xxx/3$) at the time of dividing Prefix, and divided it into every [a triplet] elongates Prefix, and it arranges an entry as required. For example, when elongating Node D (0011) to 6 bits, it will have four entries, 001100, 001101, 001110, and 001111.

[0005] Supposing retrieval of a Multiway tree gives a maximum of two or more E child nodes (≥ 3 and drawing E= 8) to one parent node Compare even the high order Prefix length bit eye of Prefix and an IP address which a node has, and when in agreement It will move, if it sees [the 1st / +/ bit of from the high order Prefix length of an IP address to / high order Prefix length + $\log_2 E$ bit] and there is a corresponding child node, and retrieval is continued. Next which the table corresponding to the node which ended retrieval and was in agreement immediately before when there was no corresponding child node, and when a node compared and it was not in agreement has Retrieval is ended as information which searches for Hop information. Since the same Prefix was expressed with this method, more two or more nodes may be needed, and there was a problem of wasting memory (node D in drawing 4 etc.). Although the count of access can be reduced so that the number of child nodes which one parent node has is increased If the number of bits (it is 128 bits at the time of 32 bits and IPv6 at the time of IPv4) assigned to an IP address increases When the wooden depth became deep, the required amount of memory increased and the realistic amount of memory was considered, there was a problem of it becoming impossible to perform effectively that it becomes impossible for one parent node to have not much many child nodes, and it decreases the count of access.

[0006]

[Problem(s) to be Solved by the Invention] thus, by the routing table searching method using CAM used conventionally Since it is necessary to store the Prefix [all] itself in CAM, there is a problem that many amounts of memory are needed, and by the routing table searching method using a PATORISHIA tree Since the count of retrieval increases the more the more the wooden depth becomes deep, there is a problem that retrieval takes time amount and by the routing table searching method using Multiway further If the number of bits assigned to an IP address increased, since the wooden depth would become deep, the required amount of memory increased, it becomes impossible to have had many child nodes, and there was a problem of it becoming impossible to fully reduce the count of access.

[0007] Then, the purpose of this invention solves the technical problem of these former, decreases the count of retrieval required for retrieval and the amount of memory of routing table, and is to offer a high speed and the mass routing table searching method.

[0008]

[Means for Solving the Problem] In order to attain the above-mentioned purpose, by the routing table searching method of this invention, retrieval is divided into two steps and performed, in the 1st step of processing, the 1st search is performed using CAM, and a search is performed within limits which arrange a tree to the 2nd step and by which the tree was limited to it in the 2nd retrieval. Therefore, Prefix which divided all Prefix(es) by which the entry is carried out to routing table into the range determined by Prefix length, respectively, and shortened divided Prefix to the smallest Prefix length in the range is stored in the 1st step of CAM, and the destination IP address which should be searched in case retrieval is started is inputted into CAM. Prefix which compares a part for the high order Prefix length bit of a destination IP address inputted as Prefix stored in CAM if the destination IP address which should be searched was inputted, and has the longest Prefix length among congruous Prefix(es) is made into an answer, and the address of memory with which the 2nd step of tree corresponding to Prefix

calculated as an answer is stored is outputted from the 1st step. In the 2nd step of processing, the destination IP address which should be searched, and the address outputted from the 1st step are read. From the address outputted from the 1st step, the node which exists in the middle of the tree corresponding to the address A tree is searched within the limits of the Prefix length of the range where Prefix which read from the memory which stores the 2nd step of tree, started retrieval from the node, performed a search in the Ruhr decided to be the tree, and was in agreement by the 1st step of CAM was contained. It is Next when retrieval is completed in the Ruhr decided to be a tree. Information or Next of Hop The index information on other tables with the information on Hop is outputted as a final result. [0009] In this invention, Prefix which divided all Prefix(es) by which the entry is carried out to routing table into the range determined by Prefix length, and shortened divided Prefix to the smallest Prefix length in the range is stored in CAM. Consequently, Prefix used as the answer obtained in the 1st step using CAM In order to be referred to as Prefix with the longest Prefix length in Prefix which compares the inputted IP address with Prefix stored in CAM, and is in agreement with the inputted IP address, Since a longest match Prefix cannot exist even if it searches retrieval by the 2nd step of tree with Prefix length longer than the longest Prefix length in the range of Prefix length where the Prefix was contained, It can ask for a longest match Prefix by searching only in the range of Prefix length in which the Prefix was contained.

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-358064

(P2000-358064A)

(43) 公開日 平成12年12月26日 (2000. 12. 26)

(51) Int.Cl.⁷

H 0 4 L 12/56

識別記号

F I

H 0 4 L 11/20

ターミナル* (参考)

1 0 2 D 5 K 0 3 0

審査請求 未請求 請求項の数 3 O L (全 9 頁)

(21) 出願番号

特願平11-170479

(22) 出願日

平成11年6月17日 (1999. 6. 17)

(71) 出願人 000004226

日本電信電話株式会社

東京都千代田区大手町二丁目3番1号

(72) 発明者 宇賀 雅則

東京都新宿区西新宿三丁目19番2号 日本

電信電話株式会社内

(72) 発明者 塩本 公平

東京都新宿区西新宿三丁目19番2号 日本

電信電話株式会社内

(74) 代理人 100077274

弁理士 磯村 雅俊 (外1名)

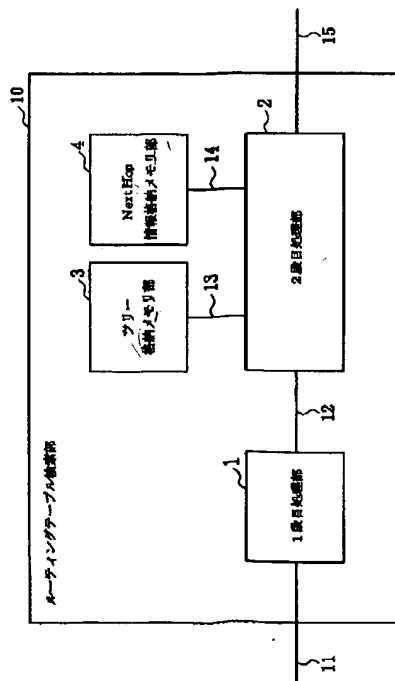
Fターム(参考) 5K030 GA01 GA06 KA02 KA05 LB05

(54) 【発明の名称】 ルーティングテーブル検索装置および検索法

(57) 【要約】

【課題】ルーティングテーブル検索の回数とメモリ量を低減し、高速検索する。

【解決手段】検索を2段に分けて行い、1段目処理部1でCAMを用いて1回目の検索を行い、2段目処理部2にツリーを配置して2回目の検索を行う。テーブルにエントリされている全てのPrefixをPrefix長で決められた範囲に分け、そのPrefixをその範囲中で最小のPrefix長に短縮したPrefixをCAMに格納する。1段目処理部1では入力IPアドレスとCAMに格納されたPrefixを比較し、一致したときそのPrefixに対応するツリー格納メモリ部3のアドレスを出力する。ツリーの途中にあるCAMの出力に該当するノードに移り、限定された範囲内で検索し、検索終了時にNextHop情報格納メモリ4の情報を最終的結果として出力する。



【特許請求の範囲】

【請求項1】 IPパケットの宛先IPアドレスから、ルーティングテーブル内のPrefixの中で最長一致のPrefixを検索し、Next Hop情報あるいはNext Hop情報を持つ他テーブルへのIndex情報を求めるルーティングテーブル検索装置において、

検索すべき宛先IPアドレスをContent Addressable Memory (CAM) に入力すると、該CAMに格納されているPrefixと前記宛先IPアドレスの上位Prefix長ビット分とを比較し、一致したPrefixのうち最長のPrefix長を持つPrefixを答えとして、2段目処理部に移る1段目処理部と、

ツリーが配置され、該ツリーの途中にあり、前記1段目処理部で一致したCAMの出力に該当するノードに検索処理が移ると、該ツリーの限定された範囲内で検索を行い、検索が終了した時点でNext Hop情報あるいはNext Hop情報を持った他テーブルへのIndex情報を最終結果として出力する2段目処理部とを具備したことを特徴とするルーティングテーブル検索装置。

【請求項2】 IPパケットの宛先IPアドレスから、ルーティングテーブル内のPrefixの中で最長一致のPrefixを検索し、Next Hop情報あるいはNext Hop情報を持つ他テーブルへのIndex情報を求めるルーティングテーブル検索法において、検索を2段に分けて行い、1段目の処理では、ルーティングテーブルにエントリされている全PrefixをPrefix長により決められた範囲にそれぞれ分け、分けられたPrefixをその範囲の中で最も小さいPrefix長へ短縮したPrefixをCAMに格納し、検索すべき宛先IPアドレスをCAMに入力し、該宛先IPアドレスの上位Prefix長ビット分とCAMに格納されているPrefixとを比較し、比較の結果、一致したPrefixのうち最も長いPrefix長を持つPrefixを答えとし、そのPrefixに対応するツリーのノードを格納しているメモリのアドレスを1段目処理部から出力することを特徴とするルーティングテーブル検索法。

【請求項3】 請求項2に記載のルーティングテーブル検索法において、

前記2段の検索の中の2段目の処理では、検索すべき宛先IPアドレスと1段目処理部から出力されたアドレスを読み込み、

1段目処理部から出力された該アドレスに対応するツリーの途中にあるノードの情報を、2段目処理部のツリーを格納しているメモリから読み込み、

該ノードから検索を開始し、ツリーに決められたルールに従って検索を行い、

1段目処理部のCAMで一致したPrefixが含まれていた範囲のPrefix長の範囲内で該ツリーを検索し、

該ツリーの検索終了時点で、Next Hop情報あるいはNext Hop情報を持つ他テーブルへのIndex情報を最終結果として出力することを特徴とするルーティングテーブル検索法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、高速かつ大容量のルーティングテーブル検索法に関し、特に検索回数を制限することができ、高速にルーティングテーブルを検索することが可能なルーティング検索法に関する。

【0002】

【従来の技術】 従来のCAM (Content Addressable Memory) を用いたルーティングテーブル検索法について説明する。図2は、CAMを用いた従来のルーティングテーブル検索法の動作説明図である。図2に示すように、Prefix長毎にCAM 16-1~16-25を用意して、ルーティングテーブル上のPrefixをそのPrefix長と対応するCAMに格納する。すなわち、Prefix長8のPrefixであればCAM (8) 16-1に格納し、Prefix長9のPrefixであればCAM (9) 16-2に格納し、同じようにしてPrefix長32のPrefixであればCAM (32) 16-25に格納する。検索を行う際には、検索すべき宛先IPアドレスを全CAM 16-1~16-25に入力する。Prefix長pに対応するCAMに入力された宛先IPアドレスは、宛先IPアドレスの上位pビット分とCAMが格納しているPrefixとを比較し、一致した場合、CAMはNext Hopの情報あるいはNext Hopの情報をを持った他テーブルへのIndex情報を出力する。なお、複数のCAMで一致した場合には、Priority Encoder 17で最も長いPrefixを格納しているCAMからの出力を選択する。図2の方式では、ルーティングテーブルにあるPrefix自身を全てCAMに格納しなければならない、その結果、多くのメモリ量が必要になるという問題があった。

【0003】 図3は、パトリシアツリーを用いた従来のルーティングテーブル検索法の説明図である。図3に示す検索法では、各ノード31~34はそれぞれPrefixとそのノードのPrefix長を持っている。あるノードに検索すべきIPアドレスが到達した場合、そのノードが示すPrefixと宛先IPアドレスの上位Prefix長ビット目までを比較する。比較の結果、一致しない場合には、直前に一致したノードに対応するテーブルが持つNext Hopの情報を求める情報として検索を終了し、一致した場合には、そのノードに対応するテーブルが持つNext Hopの情報を読み込

み、記録する。次に、宛先IPアドレスの上位Prefix長+1ビット目を参照し、その値により子ノードが存在する場合には子ノードに移り、検索を続行し、一方、子ノードが存在しない場合には、検索を終了し、記録したNext Hopの情報を求める情報とする。図3では、例えば、10000101・・・という宛先IPアドレスが入力されたとき、先ずルートノード31で宛先IPアドレスの上位1ビット目の値が0または1で分岐を行い、検査ビット（宛先IPアドレスの上位1ビット目）が1であるためノード（S）32に移り、上位3ビット分を比較して一致しているので、検査ビットである4ビット目を調べる。検査ビットが0であるためノード（U）33に移る。ここでは上位5ビット分を比較して一致しないので、検索を終了し、直前で一致したノード（S）32の100/3を答えとする。図3の方式では、ルーティングテーブルのエントリ数の増加やアドレス長が長くなることによって木の深さが深くなり、それに伴い検索回数が多くなり、その結果、検索に時間がかかるという問題があった。

【0004】図4は、Multiwayツリーを用いたルーティングテーブル検索法の説明図である。この方式は、検索の際にノードへのアクセス回数を減らす目的で、1つの親ノードに3つ以上の複数の子ノードを持たせることによって木の深さを浅くする方法である。すなわち、図4（a）から図4（b）のように変形させたツリーにすることにより、木の深さを浅くできる。図4（b）は、3ビットずつにPrefixを区切った場合の具体例（xxx/3）であって、区切った段（Prefix長が3の倍数の段）上に存在しない有効ノード（図中でA、B、E、F、D）はPrefixを伸ばして必要なだけエントリを配置する。例えばノードD（0011）を6ビットへ伸ばする場合、001100、001101、001110、001111の4つのエントリを持つことになる。

【0005】Multiwayツリーの検索は、例えば1つの親ノードに最大E個（ ≥ 3 、図ではE=8）の複数の子ノードを持たせるとすると、ノードが持つPrefixとIPアドレスの上位Prefix長ビット目までを比較し、一致した場合には、IPアドレスの上位Prefix長+1ビット目から上位Prefix長+log₂Eビット目までを見て、対応する子ノードがあれば移り、検索を続行する。対応する子ノードがなかった場合、またノードで比較した時に一致しなかった場合には、検索を終了し、直前に一致したノードに対応するテーブルが持つNext Hop情報を求める情報として検索を終了する。この方式では、1つの同じPrefixを表わすために、複数のノードがより多く必要となる場合があり（図4におけるノードD等）、メモリを浪費してしまうという問題があった。1つの親ノードが持つ子ノードの数を増やすほどアクセス回数を減らすことが

できるが、IPアドレスに割り当てられるビット数（IPv4の時は32ビット、IPv6の時は128ビット）が増加すると、木の深さが深くなり、必要なメモリ量が増加し、現実的なメモリ量を考えると、1つの親ノードがあまり多くの子ノードを持つことができなくなり、アクセス回数を減少させることが効果的に行えなくなる、という問題があった。

【0006】

【発明が解決しようとする課題】このように、従来用いられていたCAMを用いたルーティングテーブル検索法では、Prefix自身を全てCAMに格納する必要があるため、メモリ量が多く必要になるという問題があり、またバトリシアツリーを用いたルーティングテーブル検索法では、木の深さが深くなればなるほど検索回数が多くなるため、検索に時間がかかるという問題があり、さらにMultiwayを用いたルーティングテーブル検索法では、IPアドレスに割り当てられるビット数が増えると、木の深さが深くなるため、必要なメモリ量が増加し、多くの子ノードを持てなくなり、アクセス回数を十分に減らすことができなくなるという問題があった。

【0007】そこで、本発明の目的は、これら従来の課題を解決し、ルーティングテーブルの検索に必要な検索回数およびメモリ量を減少させ、高速かつ大容量のルーティングテーブル検索法を提供することにある。

【0008】

【課題を解決するための手段】上記目的を達成するため、本発明のルーティングテーブル検索法では、検索を2段に分けて行い、1段目の処理ではCAMを用いて1回目の検索を実行し、2段目にツリーを配置して2回目の検索をツリーの限定された範囲内で検索を実行する。そのために、ルーティングテーブルにエントリされている全てのPrefixをPrefix長により決められた範囲にそれぞれ分け、分けられたPrefixをその範囲の中で最も小さいPrefix長へ短縮したPrefixを1段目のCAMに格納して、検索を開始する際に検索すべき宛先IPアドレスをCAMに入力する。検索すべき宛先IPアドレスが入力されたならば、CAMに格納されているPrefixと入力された宛先IPアドレスの上位Prefix長ビット分とを比較し、一致したPrefixのうち最も長いPrefix長を持つPrefixを答えとし、答えとして求めたPrefixに対応する2段目のツリーが格納されているメモリのアドレスを1段目から出力する。2段目の処理では、検索すべき宛先IPアドレスと1段目から出力されたアドレスを読み込み、1段目から出力されたアドレスから、そのアドレスに対応するツリーの途中にあるノードを、2段目のツリーを格納しているメモリから読み込み、そのノードから検索を開始し、そのツリーに決められたルールで検索を実行し、1段目のCAMで一致したPre

fixが含まれていた範囲のPrefix長の範囲内でツリーを検索する。ツリーに決められたルールで検索が終了した時、Next Hopの情報あるいはNext Hopの情報を持つ他のテーブルへのindex情報を最終的な結果として出力する。

【0009】本発明においては、ルーティングテーブルにエントリされている全てのPrefixをPrefix長により決められた範囲に分け、分けられたPrefixをその範囲の中で最も小さいPrefix長へ短縮したPrefixをCAMに格納する。その結果、1段目20でCAMを用いて得られる答えとなるPrefixは、入力されたIPアドレスとCAMに格納されているPrefixとを比較し、入力されたIPアドレスと一致するPrefixの中で最も長いPrefix長を持つPrefixとするため、2段目のツリーでの検索は、そのPrefixが含まれていたPrefix長の範囲の中で一番長いPrefix長よりも長いPrefix長で検索しても最長一致Prefixが存在する可能性がないため、そのPrefixが含まれていたPrefix長の範囲でのみ検索を行うことで、最長一致Prefixを求めることができる。検索は、ツリーの最上段にあるノードから常に検索を開始する必要はなく、ツリーの途中のノードから検索を開始し、限られた範囲内で検索を行うことにより、最長一致Prefixを求めることができる。

【0010】

【発明の実施の形態】以下、本発明の実施例を、図面により詳細に説明する。図1は、本発明の一実施例を示すルーティングテーブル検索装置のブロック図である。図1において、0はルーティングテーブル検索部、1はCAMを用いた検索を実施する1段目処理部、2はツリーの限定された範囲内で検索を実施する2段目処理部、3はツリー格納メモリ部、4はNext Hopの情報格納メモリ部、11は入力線、15は出力線、12、13、14は信号線である。入力線11は、1段目処理部1と接続し、検索すべき宛先IPアドレスを1段目処理部1に入力する。信号線12は1段目処理部1と2段目処理部2を接続し、信号線13は2段目処理部2とツリー格納メモリ部3を接続し、信号線14は2段目処理部2とNext Hopの情報格納メモリ部4とを接続する。出力線15は、2段目処理部2からNext Hopの情報を外部に出力する。

【0011】検索が開始されると、まず入力線11から1段目処理部1へ検索すべき宛先IPアドレスを入力し、1段目処理部1で最初の検索を行い、その結果を信号線12を介して2段目処理部2へ送る。その結果はツリー格納メモリ部3のアドレスを示しており、この結果とツリー格納メモリ部3に格納されているPrefixとツリー情報を用いてツリー上で検索を行う。ツリーを用いた検索の結果は、Next Hopの情報格納メモリ部4のアドレスを示しており、2段目処理部2はこの結果をもとにしてNext Hopの情報格納メモリ部4へ信号線14を介してアクセスし、Next Hopの情報を読み込み、出力線15を介して出力する。

【0012】図5は、図1における1段目処理部の詳細ブロック図である。図5において、22はCAM処理部、20-1~20-NはそれぞれCAM処理部22内に設けたCAM、21はPriority Encoder、11は入力線、30-1~30-Nは各CAM20-1~20-Nに入力する信号線、31-1~31-Nは各CAM20-1~20-Nから出力されて、Priority Encoder21に入力される信号線、32は入力線11から直接Priority Encoder21に接続する信号線である。CAM処理部22は、入力線11から入力されたIPアドレスをCAM20-1~20-Nの全てに入力し、信号線31-1~31-Nを介してCAM20-1~20-Nからの出力をPriority Encoder21へ送出する。また、入力線11から入力されたIPアドレスは、信号線32を介してPriority Encoder21に入力される。

【0013】ルーティングテーブルにエントリされている全てのPrefixを、Prefix長により決められた範囲にそれぞれ分け、分けられたPrefixをその範囲の中で最も小さいPrefix長へ短縮したPrefixをCAM20-1~20-Nに格納している。CAM20-1~20-Nは、入力線11から入力された検索すべきIPアドレスと格納しているPrefixとを比較して、一致しているPrefixが存在した場合には、そのPrefixに対応するツリー格納メモリ部3のアドレスを信号線31-1~31-Nを介して出力する。Priority Encoder21は、信号線32を介して入力された検索すべきIPアドレスと、複数のCAMから一致したことを示す信号を受けた場合、それらのCAMの中で最も長いPrefix長を持つCAMからの出力を選択し、その出力を信号線12を介して出力する。また、一致を示す信号をどのCAMからも受けなかった場合には、信号線12を介してツリーの最も上に位置するノードに対応するアドレスを出力する。

【0014】図6は、図5におけるCAM処理部の詳細ブロックの例を示す図である。例えば、IPv4(アドレスは32ビット)のルーティングテーブルのPrefixをCAMに格納する場合、PrefixをPrefix長で3つの区切り、Prefix長8~15、Prefix長16~23、Prefix長24~32に分けたとする。その場合、図6のCAM20-1、20-2、20-3に示すように、それぞれの範囲のPrefixを格納するために3つのCAMを用意し、CAM20-1、20-2、20-3の中には、分けられた範囲

の中で最も小さいPrefix長へ短縮したPrefixをそれぞれ格納する。

【0015】図7は、図6における1つのCAMへ格納するPrefix例を示す図である。図6に示すように、CAM20-1には、Prefix長の範囲8~15に含まれる10. 2/15、12. 32/13、12. 4/14を、この範囲の最も小さいPrefix長8へ短縮したPrefixは10/8、12/8、12/8となり、同じPrefixはまとめて10/8、12/8を格納する。このようにして、CAM20-1には、Prefix長8~15のPrefixから、Prefix長8へ短縮したPrefixが格納される。

【0016】次に、図1における2段目処理部2について述べる。2段目処理部2におけるツリーは、様々なツリーを用いることができるが、ここではパトリシアツリーを用いた場合の実施例について説明する。ルーティングテーブル上のPrefixとCAMに格納されているPrefixからパトリシアツリーを作成し、ツリー格納メモリ部3へ入力する。CAMは、宛先IPアドレスが入力された際にCAMが保持するPrefixと比較し一致した場合には、そのPrefixに対応するツリー格納メモリ部3のアドレスを出力するようにしている。なお、CAMに格納されているPrefixをパトリシアツリーに付け加える際に、そのノードが保持するNext Hopの情報格納メモリ部のアドレスは2種類に分かれる。

【0017】そのアドレスの1つ目は、CAMが格納しているPrefixと同じPrefixがルーティングテーブル上に存在しない場合には、CAMが格納しているPrefixを持つノードの親ノードが持つNext Hopの情報格納メモリ部のアドレスをノードが保持する。もし、親ノードがNext Hopの情報格納メモリ部のアドレスを持っていない場合には、親ノードの親ノードが持つNext Hopの情報格納メモリ部のアドレスを保持する。以下、同様に親ノードの親ノードが持っていない場合等は、Next Hopの情報格納メモリ部のアドレスを保持する親ノードを探していく。そのアドレスの2つ目は、CAMが格納しているPrefixと同じPrefixがルーティングテーブル上に存在する場合には、ルーティングテーブル上に存在するPrefixのみツリー格納メモリ部3で保持し、保持されたPrefixに対応するツリー格納メモリ部3のアドレスをCAMは出力するようにしている。

【0018】図8は、パトリシアツリーとそれぞれのメモリの関係の具体例を示す説明図である。図8(a)にパトリシアツリーを示し、図8(b)~(d)にこのパトリシアツリーに対応するCAM、ツリー格納情報メモリ部3、Next Hopの情報格納メモリ部4を示している。ツリー格納メモリ部3の他情報は、ツリーを用いて検索を行うための情報が入っている。それぞれ1

0. 0/9、10. 34、12/22はルーティングテーブル上に存在するPrefixであり、これらのPrefixを持つノードはNext Hopの情報をもち、また10. 32/14を持つノードはパトリシアツリー上の中継ノードであって、Next Hopの情報を持たず、また10. 34/16は10. 34. 12/22をPrefix長16へ短縮したPrefixであり、CAMにも格納されている。10. 34/16がルーティングテーブル上に存在しないので、Next Hopの情報を持っていないからNext Hopの情報を保持する親ノードを探す。しかし、10. 34/16の親ノード(10. 32/14)がNext Hopの情報を持っていないため、その親ノード(10. 0/9)が持つNext Hopの情報を10. 34/16のNext Hopの情報とする。それ故、10. 34/16に対応するツリー格納メモリ部のNext Hop情報格納メモリ部のアドレスを10. 0/9と同じi1とする。

【0019】例えば、IPアドレス10. 34. 12. 50の検索を行う場合、まずCAMから出力されたツリー格納メモリ部3のアドレスt3を利用し、ツリー格納メモリ部3に格納されているPrefix10. 34/16と比較する。ここで、検索すべきIPアドレスの上位16ビットと10. 34/16が一致しているため、IPアドレスの上位17ビット目が0であるか、1であるかを調べる。ここでは0であるため、ツリー格納メモリ部3に格納されているツリー情報を用いて、ノード10. 34. 12/22に移る。このノードにおいても同様に比較を行う。比較の結果は一致するが、子ノードがないためにここで検索を終了する。求めたアドレスi2を用いてNext Hop情報格納メモリ部4へアクセスし、Xという情報をNext Hop情報格納メモリ部4から取り出して、これを図1の出力線15を通して出力する。

【0020】図9は、Multiwayツリーを本発明に適用する方法の具体例を示す説明図である。図9

(a)が本発明を適用していない場合のMultiwayツリーの一部であり、図9(b)が本発明を適用した場合のMultiwayツリーの一部で、図9(a)に対応するツリーである。図9(a)(b)では、4ビット分、つまり1つの親ノードから最大16個の子ノードを持たせ、CAMの区切り位置をPrefix長4とする。例えば、4ビット分の子ノードを持たせたならば、CAMの区切り位置をPrefix長12で初めて区切るというように区切り位置を4の倍数にするように位置決めする。つまり、子ノードが存在する位置でCAMの区切り位置を決める。ルーティングテーブルからMultiwayツリーを作り、CAMの区切り位置に存在する子ノードをCAMに格納する。検索は、図1に示す1段目処理部1の出力により、その結果に対応する範囲に

区切られた所に存在するノード（図9では、0011/4）から検索を行う。

【0021】本発明においては、（イ）ルーティングテーブルにエントリされている全てのPrefixをPrefix長により決められた範囲にそれぞれ分けて、分けられたPrefixをその範囲の中で最も小さいPrefix長へ短縮したPrefixをCAMに格納するため、CAMに格納すべきPrefixを少なくすることができる。また、（ロ）1段目でCAMを用いて得られる答えとなるPrefixは、入力されたIPアドレスとCAMに格納されているPrefixとを比較し、入力されたIPアドレスと一致するPrefixの中で最も長いPrefix長を持つPrefixとするため、2段目のツリーでの検索は、そのPrefixが含まれていたPrefix長の範囲よりも大きいPrefix長で検索しても、最長一致Prefixが存在する可能性がないため、ツリーの途中のノードから検索を開始し、限られた範囲内で検索を行うことで、最長一致Prefixを求めることができるので、検索回数を制限することが可能であり、その結果、高速にルーティングテーブルの検索を行える。

【0022】

【発明の効果】以上説明したように、本発明によれば、必要なCAMのメモリ容量を減少させることができ、またツリーの途中のノードから検索を開始し限られた範囲内で検索を行うことで最長一致Prefixを求めることができるので、検索回数を制限することができ、高速にルーティングテーブルの検索を実施することができる。

【図面の簡単な説明】

【図1】本発明の一実施例を示すルーティングテーブル検索装置のブロック図である。

【図2】従来におけるCAMを用いたルーティングテーブル検索法の説明図である。

【図3】従来におけるパトリシアツリーを用いたルーティングテーブル検索法の説明図である。

【図4】従来におけるMultiwayツリーを用いたルーティングテーブル検索法の説明図である。

【図5】図1における1段目処理部の詳細ブロック図である。

【図6】図5におけるCAM処理部の詳細ブロックの例を示す図である。

【図7】図6におけるCAMに格納するPrefix例を示す図である。

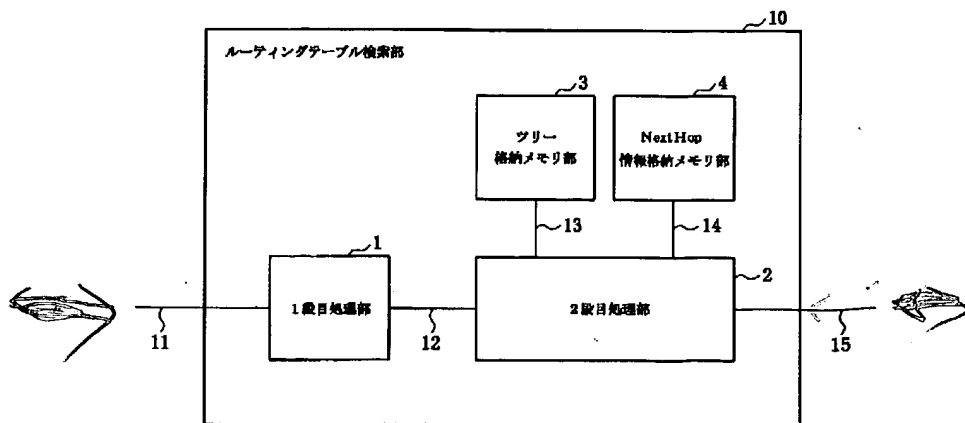
【図8】本発明のパトリシアツリーとメモリの関係を示す説明図である。

【図9】Multiwayツリーに本発明の方式を適用させた場合の例を示す説明図である。

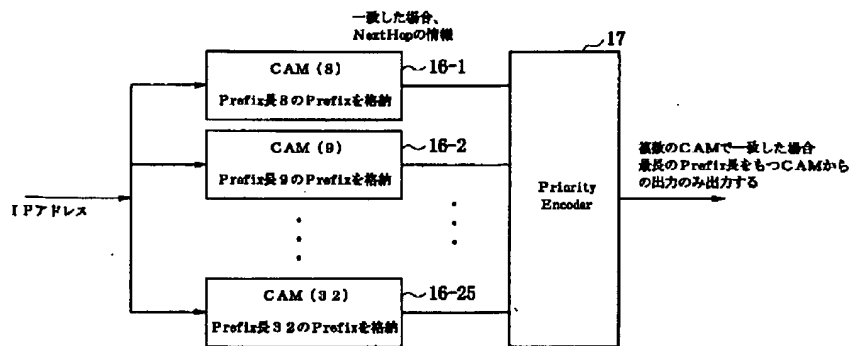
【符号の説明】

1…1段目処理部、2…2段目処理部、3…ツリー格納メモリ部、4…Next Hop 情報格納メモリ部、11…入力線、15…出力線、12, 13, 14…信号線、22…CAM処理部、16-1…Prefix長8のPrefixを格納するCAM、16-2…Prefix長9のPrefixを格納するCAM、16-25…Prefix長25のPrefixを格納するCAM、17…Priority Encoder、21…Priority Encoder、30-1~31-N…信号線、20-1…Prefix長8~15のPrefixをPrefix長8に短縮したPrefixを格納するCAM、20-2…同じくPrefix長16~23のPrefixを16に短縮したPrefixを格納するCAM、20-3…同じくPrefix長24~32のPrefixをPrefix長24へ短縮したPrefixを格納するCAM。

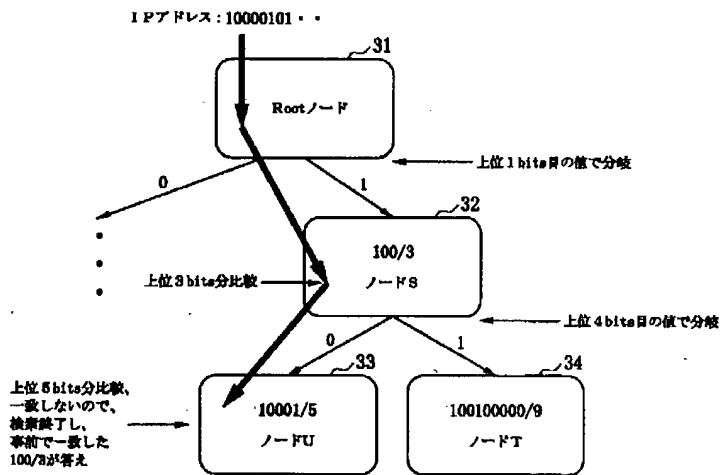
【図1】



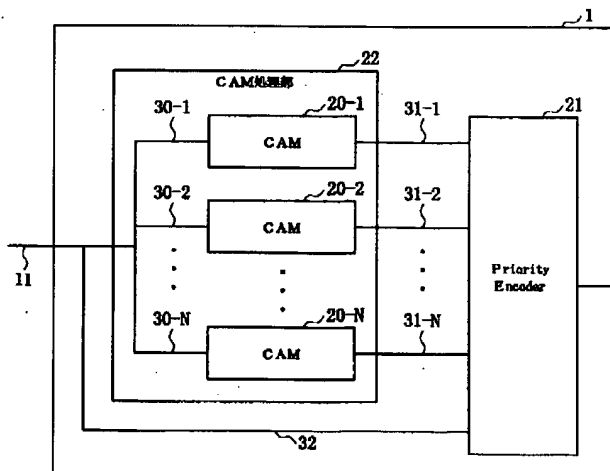
【図2】



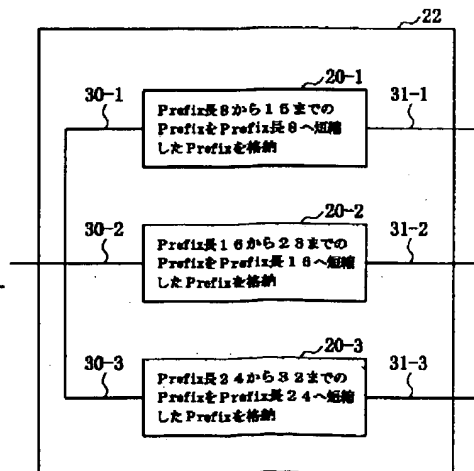
【図3】



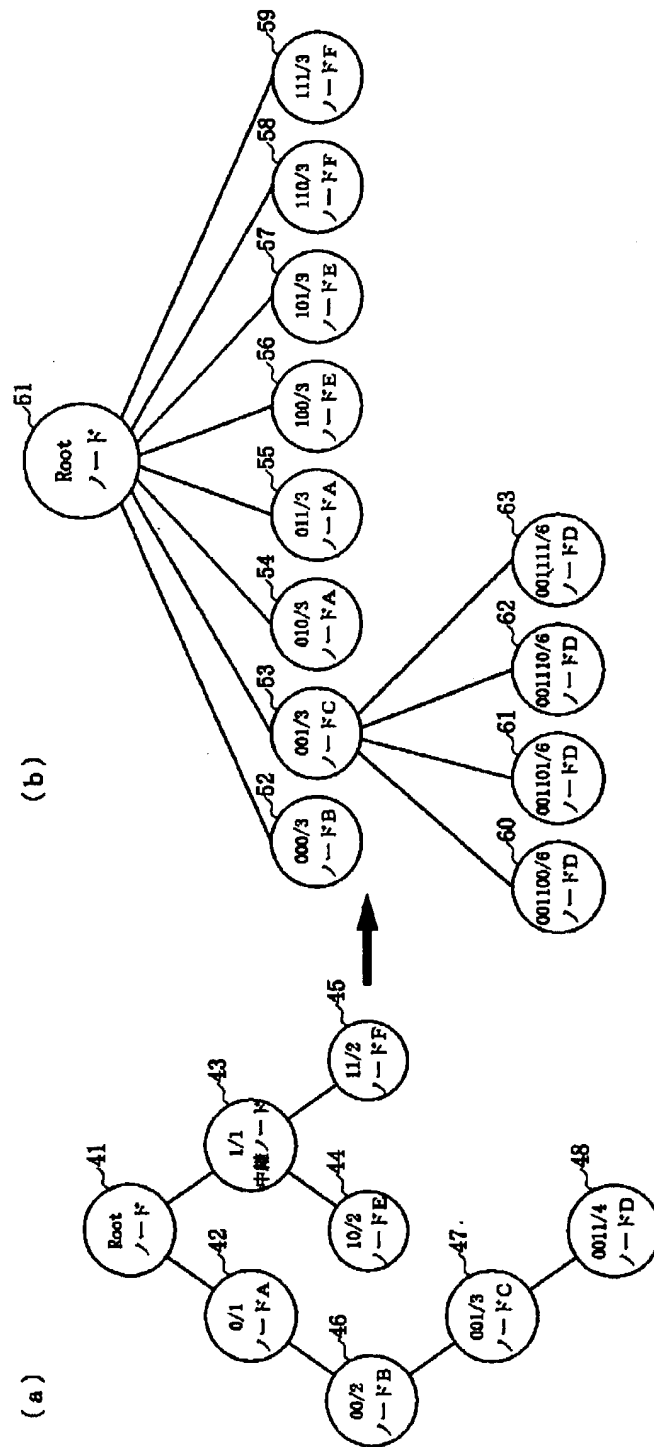
【図5】



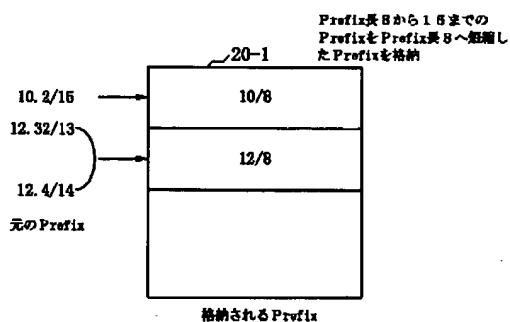
【図6】



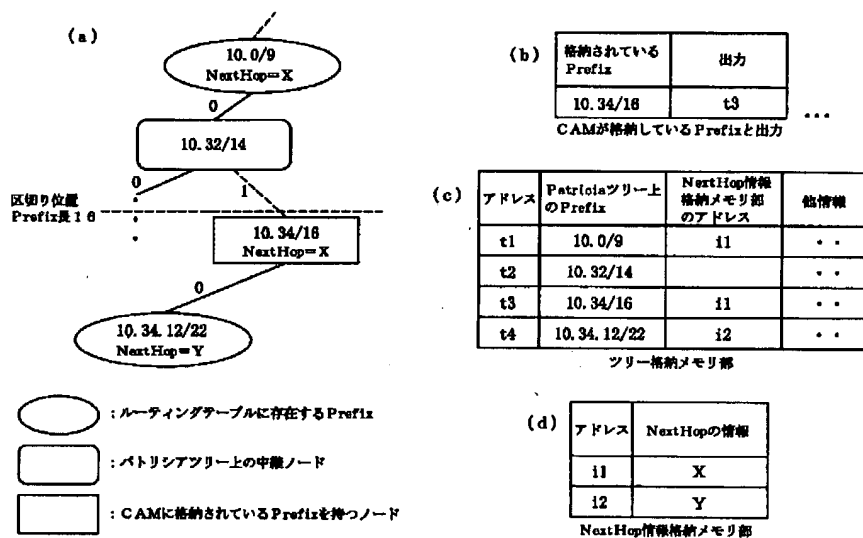
【図4】



【図7】



【図8】



【図9】

